Article:
Tallents, Gregory John orcid.org/0000-0002-1409-105X and Aslanyan, Valentin (2017) Efficient calculation of atomic rate coefficients in dense plasmas. AIP Conference Proceedings. 080001. 080001. ISSN 1551-7616

https://doi.org/10.1063/1.4975729

Efficient Calculation of Atomic Rate Coefficients in Dense Plasmas

Valentin Aslanyan[1,a)] and Greg J. Tallents[1]

[1] *York Plasma Institute, University of York, York, Heslington, YO10 5DD, United Kingdom*

[a)] Corresponding author: va567@york.ac.uk

Abstract. Modelling electron statistics in a cold, dense plasma by the Fermi-Dirac distribution leads to complications in the calculations of atomic rate coefficients. The Pauli exclusion principle slows down the rate of collisions as electrons must find unoccupied quantum states and adds a further computational cost. Methods to calculate these coefficients by direct numerical integration with a high degree of parallelism are presented. This degree of optimization allows the effects of degeneracy to be incorporated into a time-dependent collisional-radiative model. Example results from such a model are presented.

INTRODUCTION

Recent experimental studies have created plasmas at densities up to and exceeding that of solid material, by intense optical lasers[1], extreme ultraviolet[2]/x-ray free electron lasers[3] and pinches. If plasmas of this type remain cold, collisional-radiative models must take into account free electron degeneracy. Such models calculate opacities, degrees of ionization energy transport and other quantities necessary to simulate and diagnose dense plasmas. Experimental studies have recently driven interest in taking full account of the electron degeneracy in such plasmas[4, 5].

The distribution of free electrons in a plasma is governed by the statistics of fermions. For a gas of fermions at temperature T, the probability that a quantum state with energy ϵ is occupied is given by

$$F(\epsilon) = \frac{1}{\exp\left(\frac{\epsilon-\mu}{T}\right)+1}, \tag{1}$$

where μ is the chemical potential. The Fermi-Dirac energy distribution for electrons is given by $f_{FD}(\epsilon) = (G\sqrt{\epsilon}/n_e)F(\epsilon)$, where n_e is the electron density and $G = 4\pi(2m_e/h^2)^{3/2}$ (the other symbols with their usual meanings) is the degeneracy of a free electron. The chemical potential $\mu(T_e, n_e)$ is defined by the normalization condition of the distribution function: $\int_0^\infty f_{FD}(\epsilon)d\epsilon = 1$. In practice, the chemical potential may be calculated efficiently by a series expansion[6]. The Maxwell-Boltzmann distribution, used in most plasma models, is a high-temperature asymptote for the Fermi-Dirac distribution; these two are compared in Figure 1.

The effect of the Pauli exclusion principle must also be taken into account in the calculation of atomic rates. A collision involving a change of energy by a fermion requires that the final energy state be unoccupied. Therefore, for each outgoing electron the integrals for the rates of atomic processes must include a Pauli blocking factor,

$$\tilde{F}(\epsilon) = 1 - F(\epsilon). \tag{2}$$

DEFINITION OF RATE COEFFICIENTS

The rates of atomic processes are calculated by the usual integrals over a product of the energy distribution function, speed and appropriate cross section, with the addition of a Pauli blocking factor for every outgoing electron. The Fermi-Dirac distribution is mathematically more complicated than the Maxwell-Boltzmann and procludes an analytic solution to such integrals.

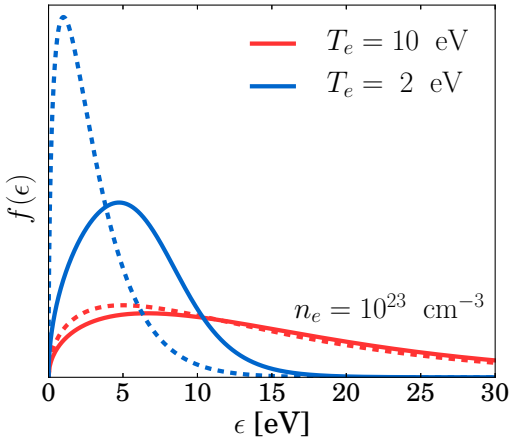

FIGURE 1. Comparison of the Fermi-Dirac distribution (solid) to the Maxwell-Boltzmann (dashed) for two temperatures, as indicated, for a free electron density $n_e = 10^{23}$ cm^{-3}.

The rate of collisional excitation between two atomic levels is given by

$$J^{\uparrow}_{FD}(E_j, T_e) = N_i G \sqrt{\frac{2}{m_e}} \int_{E_j}^{\infty} \Omega\left(\frac{\epsilon}{E_j}\right) F(\epsilon) \tilde{F}(\epsilon - E_j) d\epsilon, \tag{3}$$

where E_j is the energy separating the two atomic levels, N_i is the density of ions in the initial state and Ω the collision strength. Micro-reversibility relations may be applied to calculate the inverse rate,

$$J^{\downarrow} = \frac{g_j}{g_{j'}} \exp(E_j/T_e) J^{\uparrow}, \tag{4}$$

where g denotes the degeneracy of the respective levels.

The process of collisional ionization involves a pair of indistinguishable outgoing electrons; the total available energy left over after the collision is distributed among these electrons according to the appropriate differential cross section. The rate of collisional ionization is given by

$$K^{\uparrow}_{FD} = N_i G \sqrt{\frac{2}{m_e}} \int_{E_i}^{\infty} \int_{0}^{\epsilon_0 - E_i} \epsilon_0 \frac{d\sigma^{\uparrow}}{d\epsilon_1} F(\epsilon_0) \tilde{F}(\epsilon_1) \tilde{F}(\epsilon_0 - \epsilon_1 - E_i) d\epsilon_0 d\epsilon_1, \tag{5}$$

where E_i is the ionization energy and $d\sigma/d\epsilon_1$ the differential cross section. The differential cross section for collisional ionization is poorly studied experimentally and theoretically; Pauli blocking factors are not normally included in calculations and the differential cross section may be integrated to the total cross section through the relation $\sigma = \int_0^{\epsilon_0 - E_i} \frac{d\sigma}{d\epsilon_1} d\epsilon_1$. One experimentally verified form for the differential cross section is due to Mott[7], but it is inconsistent with more recent forms of the total cross section, such as those due to Lotz[8]. Previously, we have suggested[5] a modification to make it consistent with the Lotz and BELI[9] cross sections. The rate of the inverse process, three-body recombination, is given by

$$K^{\downarrow} = \frac{g_i}{g_{i+1}} \exp(\mu/T_e) \exp\left(\frac{E_i}{T_e}\right) K^{\uparrow}. \tag{6}$$

The rate of photoionization is largely unaffected by electron degeneracy, except for a blocking factor for the outgoing electron, so that the rate of photoionization is given by

$$L^{\uparrow}_{FD} = N_i \int_{E_i}^{\infty} \sigma_\gamma(\epsilon_\gamma) f_\gamma(\epsilon_\gamma) \tilde{F}(\epsilon_\gamma - E_i) d\epsilon_\gamma, \tag{7}$$

where $\sigma_\gamma(\epsilon_\gamma)$ is the photoionization cross section and $f_\gamma(\epsilon_\gamma)$ is the distribution of photons. The present blocking factor does not make a significant difference for high-energy photons, but can lead to a significant drop in photoionization

and therefore opacity if the photon energy is close to the ionization potential[5]. The rate of the inverse process, radiative recombination, is negligible compared to three-body recombination at high densities and is not considered here.

ALGORITHMS FOR RATE CALCULATION

As the integrals in Equation 3, 5 and 7 cannot be solved analytically, they must be evaluated by numerical quadrature sufficiently rapidly to make possible large-scale inline calculations. The infinite uppper limit of integration poses a challenge for numerical quadrature. The integrals may be truncated at some finite upper value and the "tail" discarded, so that the resultant finite region of integration contains the integral's dominant portion. A change of variables of the form $x \propto 1/\epsilon_0$ can be used to shift the limits to 0 and 1, which guarantees fixed finite integral limits. However, in this case the bulk contribution is often localised to a small sub-region making integration over this entire region counter-productive. For example, if the region from $x = 0$ to $x = 0.9$ contributes a fraction of only 10^{-6} to the total integral, truncation is required to avoid spending calculation time on this region.

Numerical quadrature replaces a continuous integral with a finite sum of the form

$$\int_a^b f(x)dx \approx \sum_{j=1}^{N} w_j f(x_j), \tag{8}$$

where x_j are the nodes and w_j the weights of the quadrature method. The computational complexity of numerical integration is shared between the calculation of an efficient set of nodes and weights and the evaluation of the integrand[10]. For example, the Newton-Cotes methods (such as the trapezoidal) require many, but straightforwardly regularly spaced nodes to achieve convergence. The large number of atomic transitions in a typical model justifies the initial computational investment into a more efficient quadrature method, such as the Gaussian. The Golub-Welsch algorithm[11] may be used to calculate the nodes and weights in this case.

Collisional-radiative models require the repeated population of the rate matrix, which depends upon only a few variables such as the temperature and density; crucially, the elements do not depend upon one another. This allows straightforward massive parallelization, particularly suited to a Graphics Processing Unit (GPU), as shown in Figure 2. GPUs have a very high theoretical speed of computation, provided that the problem can be solved by single instruction, multiple data (SIMD) operation; naturally, repeated evaluations of an integrand function is such an example. Many GPUs have in-built capabilities for fast vector reduction (summation), so that a sum of N elements may be carried out in a time $\propto \log(N)$ instead of $\propto N$ as expected from sequential addition. The inherent latency of data transfer between GPUs and their host CPU may be overcome if the atomic model contains a sufficiently large number of transitions. Since the rate calculation for all atomic processes depends only on a small number of variables, such as density and temperature, the corresponding memory copy time is short compared to the integral calculation meaning that the calculation is largely compute bound.

TABLE 1. Comparison of the CPU and GPU calculation time per one evaluation of the double integral $K^{\uparrow}$ given in Equation (5), averaged over 1000 iterations. A single iteration involves the evaluation of 14 such double integrals, one for each excited level.

Processor	**Average computation time** [μs]
Typical single core CPU	8900
Tesla K40 GPU	154

We restrict our discussion in this work to nVidia GPUs, with threads executed in warps of 32 and code written using the CUDA API[12]. We choose the number of nodes in Equation 8 to be a multiple of 32 to both allow the same set of weights to be used for all the integrals and apportion each integral to a single warp. We compare the average time taken to calculate the collisional rates for 14 atomic levels using a GPU from nVidia's Kepler generation to a single, typical modern CPU in Table 1. This calculation has achieved 0.43 TFLOPS working in double precision, equivalent to 30% of the stated maximum 1.4 TFLOPS. A similar speedup may be achieved by parallelizing the calculation over many CPUs, which is readily achievable.

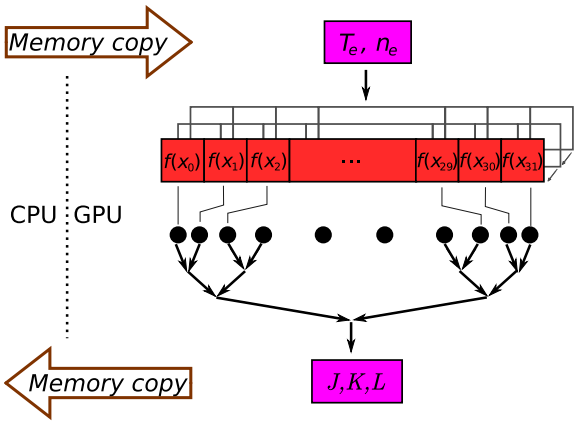

FIGURE 2. Outline for an algorithm to carry out calculation of the atomic rate coefficients by numerical quadrature on a Graphics Processing Unit (GPU). The integrand is evaluated at many nodes in parallel and summed by an efficient tree reduction operation. Integrals for each atomic level are pipelined, further increasing the parallelism. The process is efficient if the calculation is largely compute bound, *i.e.* the time taken for function evaluations is much larger than the latency of copying the integral parameters to and the results from the GPU.

EXAMPLE COLLISIONAL-RADIATIVE CALCULATION

We have assembled a simple collisonal-radiative model[13] (with present version available here[14]) in order to investigate the prospects of rapid calculation of atomic rate coefficients. We consider 14 atomic levels of four ionization stages of solid-density aluminum, from 3+ to 6+. We treat the valence electrons to be free, so lower-lying ionization stages are not present in the plasma. We consider radiation incident on the plasma to be a black-body spectrum, with a distribution given by

$$f_{\mathrm{BB}}(\epsilon_\gamma, T_r) = \frac{8\pi\epsilon_\gamma^3}{(hc)^3} \frac{1}{\exp\left(\frac{\epsilon_\gamma}{T_r}\right) - 1}, \tag{9}$$

where ϵ_γ is the photon energy and T_r the radiation temperature. Absorption by inverse bremsstrahlung is also included in the model.

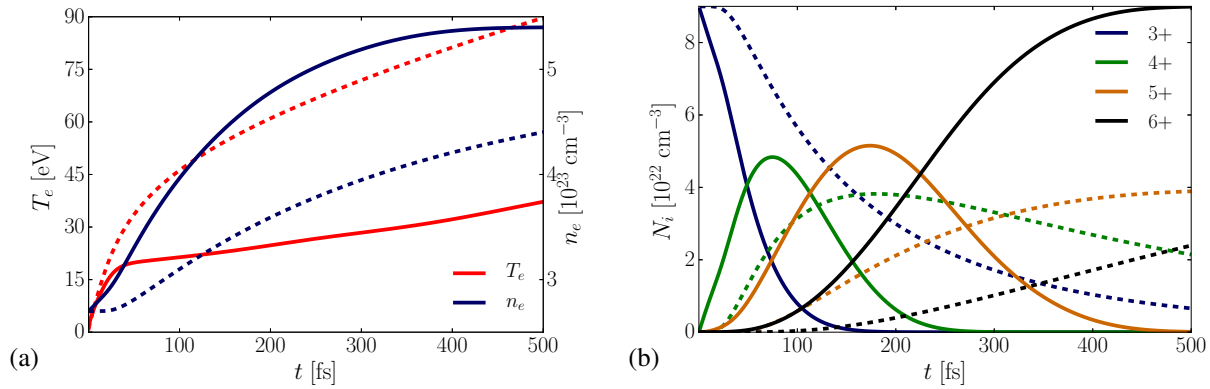

FIGURE 3. The evolution of (a) electron density n_e and temperature T_e and (b) populations of ions N_i of solid density aluminum irradiated by an isotropic black-body energy distribution with $T_r = 300$ eV for electrons modelled by Fermi-Dirac (solid) and Maxwell-Boltzmann (dashed) statistics.

The results of the simulation, with the radiation temperature $T_r = 300$ eV, are shown in Figure 3. We see a significant difference in the temporal evolution between the two models; all atomic rates in the case of the Fermi-Dirac distribution are lowered by blocking factors, but this lowering affects collisional rates more than photoionization

(compare Equations (5) and (7)) and as a result the net rate of ionization rises. This leads to a higher ionization fraction and hence higher electron density and as a result, the thermal energy of the free electrons is distributed amongst a larger population, leading to a lower final temperature. This calculation required 5×10^6 time points, taking 10 hours to complete on a single CPU and only 40 minutes to complete with the aid of a Tesla K40 GPU.

CONCLUSION

Simulations of cold, dense plasmas are greatly complicated by the effects of electron degeneracy. The computationally intensive Fermi-Dirac energy distribution takes the place of the familiar Maxwell-Boltzmann and Pauli blocking factors must be used to account for electrons undergoing a change in kinetic energy. This leads to a drop in collisional rates and absorption of radiation. We have shown that this change in rates may have a large effect on macroscopic plasma properties, such as the electron temperature, and therefore may affect large-scale plasma dynamics.

We have shown that calculations of the rates of atomic processes in degenerate plasmas may be carried out by numerical quadrature, which may be readily parallelized. In particular, GPUs may be used to achieve a significant increase in the speed of such calculations, because they are able to evaluate the integrand at many nodes simultaneously and sum the results with high efficiency. This work may help increase plasma models' utilization of modern supercomputers, which are becoming increasingly heterogeneous.

ACKNOWLEDGMENTS

The authors acknowledge UK funding from AWE Plc. and the Engineering and Physical Science Research Council (grant EP/J0194021).

REFERENCES

[1] D. J. Hoarty, S. F. James, C. R. D. Brown, B. M. Williams, H.-K. Chung, J. W. O. Harris, L. Upcraft, B. J. B. Crowley, C. C. Smith, and R. W. Lee, High Energy Density Physics **6**, p. 105 (2010).

[2] A. K. Rossall, V. Aslanyan, G. J. Tallents, I. Kuznetsov, J. J. Rocca, and C. S. Menoni, Phys. Rev. Applied **3**, p. 064013 (2015).

[3] B. I. Cho, K. Engelhorn, S. M. Vinko, H.-K. Chung, O. Ciricosta, D. S. Rackstraw, R. W. Falcone, C. R. D. Brown, T. Burian, J. Chalupsky, C. Graves, V. Hajkova, A. Higginbotham, L. Juha, J. Krzywinski, H. J. Lee, M. Messersmidt, C. Murphy, Y. Ping, N. Rohringer, A. Scherz, W. Schlotter, S. Toleikis, J. J. Turner, L. Vysin, B. W. T. Wang, D. Z. U. Zastrau, R. W. Lee, B. Nagler, J. S. Wark, and P. A. Heimann, Phys. Rev. Lett. **109**, p. 245003 (2012).

[4] B. Deschaud, O. Peyrusse, and F. B. Rosmej, EPL **108**, p. 53001 (2014).

[5] V. Aslanyan and G. J. Tallents, Phys. Rev. E **91**, p. 063106 (2015).

[6] V. V. Karasiev, D. Chakraborty, and S. B. Trickey, Computer Physics Communications **192**, p. 114 (2015).

[7] N. Oda, Radiat. Res. **64**, p. 80 (1975).

[8] W. Lotz, Z. Physik **2**, p. 232 (1970).

[9] K. L. Bell, H. B. Gilbody, J. G. Hughes, A. E. Kingston, and F. J. Smith, J. Phys. Chem. Ref. Data **12**, p. 891 (1983).

[10] L. N. Trefethen, SIAM Review **50**, p. 67 (2008).

[11] G. H. Golub and J. H. Welsch, Mathematics of Computation **23**, p. 221 (1969).

[12] E. Kandrot and J. Sanders, *CUDA by Example: An Introduction to General-Purpose GPU Programming* (Addison-Wesley, Boston, 2011).

[13] V. Aslanyan and G. J. Tallents, Phys. Plasmas **21**, p. 062702 (2014).

[14] Github repository, *https://github.com/Valentin-Aslanyan/Fermi-Dirac-GPU*.